Machine Learning in the calibration process of MercuryDPM Discrete Particle Model

Quang Hung Nguyen

Head Supervisor: Thomas Weinhart, Daily Supervisor: Anthony Thornton, Additional member: Chen Kuan.
Multi-Scale Mechanics Group, Faculty of Engineering Technology, University of Twente.

1 Introduction

Figure 1: Examples of Granular Materials.[1]

Granular material is a family of material characterized by its large bulk of densely packed particles, ranging from nanometers to centimeters [2], and is able to resist deformation and form heaps, i.e., behave like a solid and withstand strong shear force [3]. Simple examples of granular materials include sand, gravel, clays, seeds, nuts, and all ranges of powders such as coffee powder, cement powder, which is shown in figure 1. Furthermore, many processes and equipments in chemical plants use granular materials, such as catalysis, adsorption, and heat exchangers. Granular materials are projected to make about half of the products and three-quarters of the raw materials used in the chemical industry [4]. Thus, understanding how granular materials behave is of great significance.

Granular material's bulk mechanical behavior is simulated using a Discrete Particle Model (DPM, or Discrete Element Method - DEM), which generates the movement of individual particles to capture the macro-scale behavior. The DPM is a family of numerical methods for computing the motion of a large number of particles [5]. Since the properties of granular materials differ wildly, these simulations require an extensive calibration process designed individually for each type of granular material. Some parameters of the granular material model can be measured directly, such as size distribution or density. However, other parameters are effective parameters (i.e., they result from a simplified particle model) and thus cannot be directly measured. These parameters are then calibrated by choosing a few standard calibration setups (rotating drum, heap test, ring shear cell) and simulating these setups in a DPM simulation, and the missing parameters are determined such that the response of the experimental and simulation setups match.

Recently, coupled with the raise of Machine Learning in other fileds, it has also been applied to solve the calibration problem. This has been done using a Neural Network [6, 7, 8], Genetic Algorithm [9], and a recursive Bayesian sequential Monte-Carlo filtering algorithm named GrainLearning [10]. In this Assignment, two Machine Learning algorithms will be discussed: Neural Network and GrainLearning.

Artificial Neural Network (ANN) is a set of algorithms that seeks to identify correlations in data utilizing a technique that inspired by the way human brain operates - mimicking how each neurons in the brain signaling each other. The most basic ANN models is the Feedforward Multilayer Perceptron Neural Network (MLPNN), in which the purpose is to define the mapping between the input and output $y = f(x; \theta)$, and approximate the parameter θ which results in the best possible function. In MLPNN, the data will flows in one direction from the input to output, hence the name feedforward. Like other supervised learning algorithms, a MLPNN need to be trained before it can accurately describe the relations between the input and output. This is typically done by feeding the network with pre-labeled data, compare the model's output with the desired output, and update the weights parameter θ - a process called backpropagation. In the context of this Assignment, Neural Network (NN) will be used when refering to Feedforward Multilayer Perceptron Neural Network.

On the other hand, GrainLearning by H.Cheng et al, utilizes the recursive Bayesian algorithm to estimate the uncertainty parameters in DPM. Initially, a wide range of parameter space is quasi-randomly sampled numbers from the initial guess range to create a prior distribution of each parameter. Then, conditioned on the experimental values, the posterior distribution of the parameters is updated recursively by Sequential Monte-Carlo Filtering (SMC Filter) and fitted to a Gaussian Mixture Model. This process is done iteratively, until a desired value that minimises the loss function is reach, typically 3 iterations.

These two algorithms set to treat the calibration problem in two different ways, and likewise, solve it in two different ways: While GrainLearning looks to identify the microparameters from the experimental and DEM simulations's bulk parameters (inverse problem), the Neural Network will help generating a database that can map different microparameters combinations to their corresponding bulk parameters, generated by DEM simulations. In other word, NN will learn the built-in relationship between the micro- and macroparameters of the Discrete Particle Model, thus allow a much faster prediction compare to a full DEM simulation. One clear advantage of GrainLearning compares to other Machine Learning algorithms such as Neural Network is that it is an unsupervised learning algorithm, i.e., it can starts calibrating with a minimal amount of input information. A normal calibration routine, currently implemented in MercuryDPM would only need the measurements data, parameter range, and the importance weight of each measurement (depends on the modeller's knowledge). Meanwhile, the current approach mentioned in [6, 7, 8] would requires modeller to define a different NN model for each material, train it using a set of DEM simulations, and then validate the correct combinations of input-output by experiment. And although this process can be automated, to date the author has not been aware of any study implemented a fully-automated calibration routine using Neural Network.

GrainLearning has been implemented in MercuryDPM and produces satisfactory results [11]. However, one known problem to GrainLearning is that the output will only converges without reaching the "true" optimal solution.

2 Calibration

2.1 MercuryDPM

The DEM package used in this Assignment is MercuryDPM. MercuryDPM is an open-source DEM software package, developed by Thomas et al.[11].

2.2 GrainLearning

2.3 Neural Network

In order to choose a correct NN model that can describes the relationship of micro- and macroparameters as indicated in the contact law, a "bottom-up" method is employed. Initially, a simple one-input one-output NN will be built, to assess how many layers (and neurons) it would take to learn a quadratic relationship $y = x^2$, and inverse relationship $y = 1/x$.

In the next step, a slightly more complex problem is analysed: a 4-D input and 4-D output model, with contact laws described in table xxx. Finally, a fully-functional model will be coupled with DEM simulations from MercuryDPM.

Certain reccomendations will also be taken into account, (7 and 15?)

3 Experimental Section

3.1 DEM Simulations

References

[1] A. Fernandes, H. Gomes, E. M. B. Campello, P. Pimenta, A fluid-particle interaction method for the simulation of particle-laden fluid problems, 2017. doi:10.20906/CPS/CILAMCE2017-0139.

[2] A. B. Yu, F. Bassani, G. L. Liedl, P. Wyder, Powder Processing: Models and Simulations, Encyclopedia of Condensed Matter Physics, Elsevier, Oxford, 2005, pp. 401–414. doi:10.1016/B0-12-369401-9/00556-8.

[3] E. S. Oran, J. P. Boris, R. A. Meyers, Encyclopedia of Physical Science and Technology (Third Edition), Academic Press, New York, 2002, Ch. Fluid Dynamics, pp. 31–43. doi:10.1016/B0-12-227410-5/00248-9.

[4] R. M. Nedderman, Statics and Kinematics of Granular Materials, Cambridge University Press, Cambridge, 1992, Ch. Introduction, pp. 1–6. doi:10.1017/CBO9780511600043.002.

[5] X. Weng, Modeling of complex hydraulic fractures in naturally fractured formation, Journal of Unconventional Oil and Gas Resources 9 (2015) 114–135. doi:10.1016/j.juogr.2014.07.001.

[6] L. Benvenuti, C. Kloss, S. Pirker, Identification of dem simulation parameters by artificial neural networks and bulk experiments, Powder Technology 291 (2016) 456–465. doi:10.1016/j.powtec.2016.01.003.

[7] P. He, Y. Fan, B. Pan, Y. Zhu, J. Liu, D. Zhu, Calibration and verification of dynamic particle flow parameters by the back-propagation neural network based on the genetic algorithm: Recycled polyurethane powder, Materials 12 (20) (2019). doi:10.3390/ma12203350.
URL https://www.mdpi.com/1996-1944/12/20/3350

[8] D. Schiochet Nasato, R. Queiroz Albuquerque, H. Briesen, Predicting the behavior of granules of complex shapes using coarse-grained particles and artificial neural networks, Powder Technology 383 (2021) 328–335. doi:https://doi.org/10.1016/j.powtec.2021.01.029.
URL https://www.sciencedirect.com/science/article/pii/S0032591021000450

[9] H. Q. Do, A. M. Aragón, D. L. Schott, A calibration framework for discrete element model parameters using genetic algorithms, Advanced Powder Technology 29 (6) (2018) 1393–1403. doi:10.1016/j.apt.2018.03.001.

[10] H. Cheng, T. Shuku, K. Thoeni, H. Yamamoto, Probabilistic calibration of discrete element simulations using the sequential quasi-monte carlo filter, Granular Matter 20 (1) (2018) 11. doi:10.1007/s10035-017-0781-y.

[11] T. Weinhart, L. Orefice, M. Post, M. P. van Schrojenstein Lantman, I. F. Denissen, D. R. Tunuguntla, J. Tsang, H. Cheng, M. Y. Shaheen, H. Shi, P. Rapino, E. Grannonio, N. Losacco, J. Barbosa, L. Jing, J. E. Alvarez Naranjo, S. Roy, W. K. den Otter, A. R. Thornton, Fast, flexible particle simulations — an introduction to mercurydpm, Computer Physics Communications 249 (2020) 107129. doi:10.1016/j.cpc.2019.107129.